

Algebraic List Decoding of q -ary Reed-Muller and Product Reed-Solomon Codes

Nandakishore Santhi¹

¹T-13 / T-CNLS
Los Alamos National Laboratory

Postdoc Talk, Feb 15, 2007

Outline of Talk

Introduction

Some q -ary Linear Codes

The Pellikaan-Wu Algorithm

A Recursive Decoding Algorithm

Conclusions

Error Correction Codes

- ▶ A q -ary linear error-correction code $\mathbb{C}[n, k, d]$ of length n and dimension k is just a k -dimensional subspace of the n -dimensional vector space \mathbb{F}_q^n over the finite field \mathbb{F}_q . So, $\mathbb{C}[n, k, d] \subseteq \mathbb{F}_q^n$.
- ▶ The minimum *Hamming distance* d of the code is the least number of positions in which any two of its vectors differ. In typical applications, the larger the minimum distance, the better the code.
- ▶ Error correction codes are ubiquitous - they are to be found in DVDs, CDRoms, computer hardware, communication systems, space etc.

Some applications of Codes



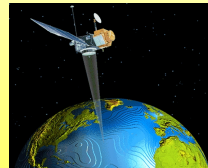
(a) CDROM



(b) Hard Disks



(c) Phones



(d) Space

Codes and Error Correction

- In typical applications, we wish to recover codewords corrupted by noise: $r = c + e$, where $c \in \mathbb{C}$.

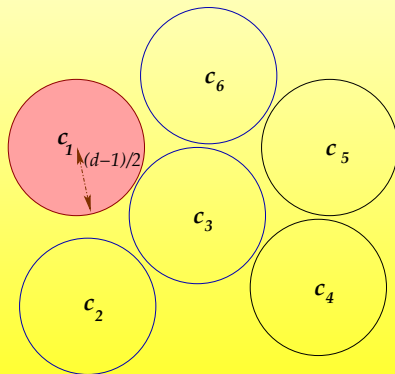


Figure: Any error of weight less than $(d-1)/2$ can be corrected.

Finite Fields

- ▶ A finite field \mathbb{F}_q is a set of q elements along with the binary operations \cdot and $+$. There are the usual properties of associativity, distributivity, identity and inverse for these operations.
- ▶ q is a prime power. So $q = p^\mu$. The prime p is called the *field characteristic*.
- ▶ There exists elements α in the field \mathbb{F}_q called *primitive elements* so that, $\mathbb{F}_q = \{0, 1, \alpha, \alpha^2, \dots, \alpha^{q-2}\}$.
- ▶ If $v > \mu$ then, $\mathbb{F}_{p^\mu} \subset \mathbb{F}_{p^v}$. Moreover, the extension field \mathbb{F}_{p^μ} can be represented as a vector space \mathbb{F}_p^μ over the base field \mathbb{F}_p .
- ▶ Let $\beta \in \mathbb{F}_q$. Then *Fermat's Little Theorem* states that $\beta^q = \beta$.

Reed-Solomon (RS) Codes

- ▶ **Reed-Solomon** codes are Maximum Distance Separable (MDS) codes. Consider $\mathcal{RS}_q(n; k; d)$. We have $d = n - k + 1$.
- ▶ Let $n \leq q$. An RS codeword can be thought of as the evaluations over the field \mathbb{F}_q of a degree $(k - 1)$ polynomial in $\mathbb{F}_q[x]$.
- ▶ Let $\mathbf{f} = [f_0 \ f_1 \ f_2 \ \dots \ f_{k-1}]$ be an information vector. Represent it as an information polynomial $f(x) = f_0 + f_1x + f_2x^2 + \dots + f_{k-1}x^{k-1}$. Let α be a primitive element of \mathbb{F}_q . Then the corresponding Reed-Solomon codeword is $[f(0) \ f(1) \ f(\alpha) \ f(\alpha^2) \ \dots \ f(\alpha^{n-2})]$.

Algebraic decoding of RS codes

- ▶ Let $\rho = k/n$ denote the *code rate*. Various algebraic RS decoders due to Berlekamp et al.[1960s] can decode *any* error pattern of weight less than $n \cdot (1 - \rho)/2$.
- ▶ Recent progress made by Sudan[1997] and Guruswami-Sudan[1999] resulted in an algebraic *list* decoder for RS codes which can correct any error pattern of weight less than $n \cdot (1 - \sqrt{\rho})$.
- ▶ Such decoders are called *bounded distance algebraic decoders*.

Berlekamp vs Guruswami-Sudan

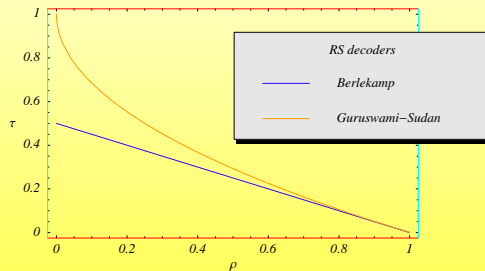


Figure: Comparison of bounded distance algebraic decoders for RS codes. τ is the relative error correction radius.

Multidimensional Extensions to RS codes

- ▶ Product Reed-Solomon (PRS) code
 - ▶ **Product Reed-Solomon** codes are natural extensions of RS codes. A codeword in the m -dimensional PRS code $\mathcal{PRS}_{q,m}(q^m; k_1, k_2, \dots, k_m; d)$ is formed from the q^m evaluations of an m -variate polynomial with degree at most $(k_j - 1)$ in variable x_j .
 - ▶ The codewords of an m -dimensional PRS code can be naturally represented over an m -dimensional hypercube.
- ▶ Reed-Muller (RM) code
 - ▶ Another multidimensional extension of a RS code is the Reed-Muller code. A codeword in the m -dimensional RM code $\mathcal{RM}_q(\ell; m; q^m)$ is formed from the q^m evaluations of an m -variate polynomial with *total degree* at most ℓ .
 - ▶ Typically a Reed-Muller code can be thought of as a sub-code of a PRS code.

A List Decoding Algorithm for RM codes

- ▶ We wish to obtain an algebraic decoding algorithm for RM codes.
- ▶ For this we will try to *embed* the m -dimensional RM code over \mathbb{F}_q inside a RS code over the extension field \mathbb{F}_{q^m} . We can then use one of the existing RS decoding algorithms.
- ▶ This strategy leads us to a form of the Pellikaan-Wu[2005] list decoding algorithm for RM codes. So what do we gain?
- ▶ We get an easily accessible correctness proof for the Pellikaan-Wu algorithm using only basic notions from Galois theory.
- ▶ As an added benefit we get a constructive proof for the famous *DeMillo-Lipton-Schwartz-Zippel* lemma. This lemma is an upper bound on the number of distinct zeros of a multi-variate polynomial over a finite field.

Correctness Proof (1)

Let $\alpha_j \in \mathbb{F}_q^m$. Then,

$$\mathcal{RM}_q(\ell, m, n) \stackrel{\text{def}}{=} \{ [\varphi(\alpha_1) \varphi(\alpha_2) \cdots \varphi(\alpha_n)] \\ | \varphi \in \mathbb{F}_q[x_1, x_2, \dots, x_m], \deg(\varphi) \leq \ell \} \quad (1)$$

- ▶ Let $\{a_1, a_2, \dots, a_m\}$ be a basis for \mathbb{F}_{q^m} over \mathbb{F}_q . For example one might as usual use a polynomial basis $\{1, \xi, \xi^2, \dots, \xi^{m-1}\}$ where ξ is any primitive element in \mathbb{F}_{q^m}
- ▶ Let $[x_1 x_2 \dots x_m] \in \mathbb{F}_q^m$.
- ▶ Then the map $\psi: \mathbb{F}_q^m \rightarrow \mathbb{F}_{q^m}$ defined as in (2) is an isomorphism.

$$[x_1 x_2 \dots x_m] \mapsto X \stackrel{\text{def}}{=} \sum_{j=1}^m a_j x_j \quad (2)$$

Correctness Proof (2)

- ▶ Claim: To obtain a decoding algorithm for RM codes, all we have to do is to form the reverse map of (2).
- ▶ Form the linear system:

$$A \cdot [x_1 \ x_2 \ \dots \ x_m]^T = [X \ X^q \ X^{q^2} \ \dots \ X^{q^{m-1}}]^T \quad (3)$$

where,

$$A \stackrel{\text{def}}{=} \begin{bmatrix} a_1 & a_2 & \dots & a_m \\ a_1^q & a_2^q & \dots & a_m^q \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{q^{m-1}} & a_2^{q^{m-1}} & \dots & a_m^{q^{m-1}} \end{bmatrix} \quad (4)$$

We used only *Fermat's little theorem* and (2) in forming A .

- ▶ So the reverse isomorphism for (2) is:

$$X \mapsto [x_1 \ x_2 \ \dots \ x_m]^T \stackrel{\text{def}}{=} A^{-1} \cdot [X \ X^q \ X^{q^2} \ \dots \ X^{q^{m-1}}]^T \quad (5)$$

Correctness Proof (3)

- ▶ For (5) to be valid, A should be invertible. Because $\{a_1, a_2, \dots, a_m\}$ is a basis for \mathbb{F}_{q^m} over \mathbb{F}_q , it follows from [1, Corollary 2.38, pp. 58] that A is non-singular.
- ▶ Now using (5) one can rewrite any m -variate polynomial in $\mathbb{F}_q[x_1, x_2, \dots, x_m]$ as a uni-variate polynomial in $\mathbb{F}_{q^m}[X]$. If the multivariate polynomial has total degree at most ℓ , the uni-variate polynomial has degree at most ℓq^{m-1} .
- ▶ So, if $\ell \leq q$ then

$$\mathcal{RM}_q(\ell, m, n) \subseteq \mathcal{RS}_{q^m}(n, \ell q^{m-1}) \cap \mathbb{F}_q^n \quad (6)$$

Pseudo Code (1)

RM-List-1

INPUT: $q, \ell \leq q, m, n \leq q^m; r = [r_1 \ r_2 \ \dots \ r_n] \in \mathbb{F}_q^n$.

STEPS:

1. Compute the parameter $t = \left\lceil n \left(1 - \sqrt{\ell q^{m-1}/n} \right) \right\rceil$.
2. Using Guruswami-Sudan algorithm find a list \mathcal{L} of codewords $c \in \mathcal{RS}_{q^m}(n, \ell q^{m-1})$ such that $d_H(c, r) < t$.
3. For every $c \in \mathcal{L}$ check if $c \in \mathbb{F}_q^n$:
 - i. If NO then discard c from \mathcal{L} .
 - ii. If YES then check if $c \in \mathcal{RM}_q(\ell, m, n)$:
 - a. If NO then discard c from \mathcal{L} .
 - b. If YES then keep c in the list \mathcal{L} .
4. return

OUTPUT: \mathcal{L}

Complexity and Consequences

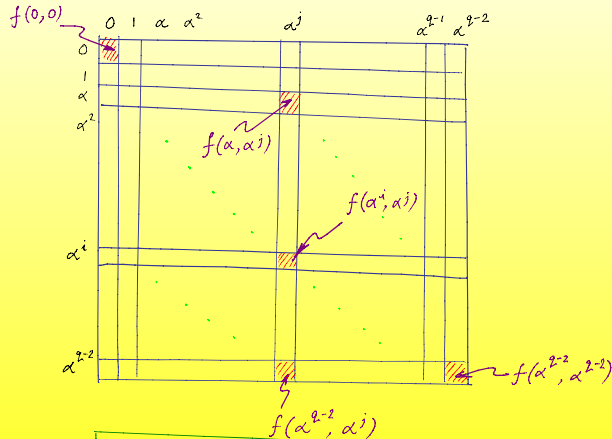
- ▶ Complexity of **RM-List-1** is $\mathcal{O}(n^3)$ field operations in \mathbb{F}_{q^m} . It can correct up to $\left\lceil n \left(1 - \sqrt{\ell q^{m-1}/n} \right) \right\rceil$ errors.
- ▶ A version of this algorithm was proposed by Pellikaan-Wu in [4]. They showed a radius of $\left\lceil n \left(1 - \sqrt{\ell(q+1)^{m-1}/n} \right) \right\rceil$.
- ▶ Product Reed-Solomon codes can be decoded using **RM-List-1**, achieving a relative error correction radius of $(1 - \sqrt{\sum_{i=1}^m \rho_i})$, where $\rho_i \stackrel{\text{def}}{=} k_i/q$.
- ▶ Our proof also showed that any non-zero multivariate polynomial $\varphi(x_1, x_2, \dots, x_m)$ of total degree ℓ has at most ℓq^{m-1} zeros in \mathbb{F}_q^m which is the *DeMillo-Lipton-Schwartz-Zippel* lemma. The original proofs use probabilistic arguments.

A Better Algebraic Decoder for q -ary PRS and RM codes

- ▶ A codeword in the code $\mathcal{PRS}_{q,m}(q^m, k_1, \dots, k_m)$ can be described within an m -dimensional cube of side length q .
- ▶ When $m = 1$, Guruswami-Sudan algorithm can decode up to $q \cdot (1 - \sqrt{k_1/q})$ errors.
- ▶ Claim: If we can correct t_{M-1} errors when $m = M - 1$, then we can correct $t_M = t_{M-1} \cdot q \cdot (1 - \sqrt{k_M/q})$ errors when $m = M$.

Description of a 2D PRS codeword.

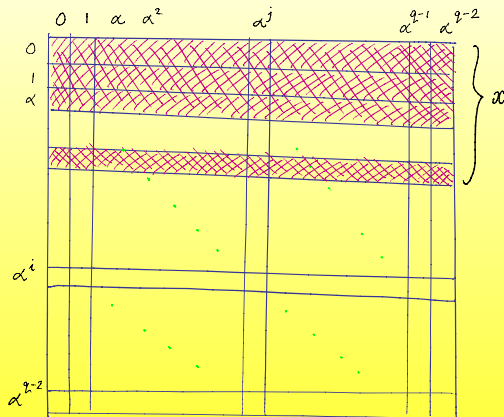
A 2D Product Reed Solomon codeword described on a square.



$$f(x_1, x_2) = \sum_{i=0}^{k_1-1} \sum_{j=0}^{k_2-1} f_{ij} x_1^i x_2^j$$

Sketch of proof - 2D case

Recursive decoding of 2D Product Reed Solomon code - all errors of weight at most $t_2 = q^2 \cdot (1 - \sqrt{\rho_1}) \cdot (1 - \sqrt{\rho_2})$ are corrected.



$$t_1 x = q \cdot (1 - \sqrt{\rho_1}) \cdot x \leq t_2 = q^2 \cdot (1 - \sqrt{\rho_1}) \cdot (1 - \sqrt{\rho_2})$$

$$\Rightarrow x \leq q \cdot (1 - \sqrt{\rho_2})$$

The recursive algorithm

- ▶ Let $\llbracket c_{i_1, i_2, \dots, i_m} \rrbracket$ be a codeword in $\mathcal{PRS}_{q,m}(q^m, k_1, \dots, k_m)$, where each of the indices i_j take values in the range $\{1, \dots, q\}$.
- ▶ $\llbracket c_{i_1, i_2, \dots, i_{j-1}}^{a_j, a_{j+1}, \dots, a_m} \rrbracket$ denotes the $(j-1)$ -dimensional vector formed out of $\llbracket c_{i_1, i_2, \dots, i_m} \rrbracket$ when the coordinates indexed by $(i_j, i_{j+1}, \dots, i_m)$ are fixed at $(a_j, a_{j+1}, \dots, a_m)$ and the rest of the indices are free.
- ▶ $\llbracket c_{i_1, i_2, \dots, i_{j-1}}^{a_j, a_{j+1}, \dots, a_m} \rrbracket$ belongs to $\mathcal{PRS}_{q,j-1}(q^{j-1}, k_1, \dots, k_{j-1})$.

Pseudo Code (2)

PRS-Decoder

INPUT: $q, (k_1, k_2, \dots, k_m) : k_i < q, m; r \in \mathbb{F}_q^n$, where $r \stackrel{\text{def}}{=} \langle \langle r_{i_1, i_2, \dots, i_m} \rangle \rangle; 1 \leq i_j \leq q$.

STEPS:

1. If $m = 1$ do:
 - i. Compute the parameter $t_1 = \lceil q(1 - \sqrt{k_1/q}) \rceil$.
 - ii. Using Guruswami-Sudan algorithm find a list \mathcal{L}_1 of codewords $c_1 \in \mathcal{RS}_q(q, k_1)$ such that $d_H(c_1, \langle r_{i_1} \rangle) < t_1$.
 - iii. Search \mathcal{L}_1 for c_1 such that $d_H(c_1, \langle r_{i_1} \rangle)$ is least. Substitute in-place the positions corresponding to $\langle r_{i_1} \rangle$ in r with c_1 and **return**.
2. For $a_m = 1, 2, \dots, q$ do:
 - i. Set $r' \leftarrow \langle r_{i_1, i_2, \dots, i_{m-1}}^{a_m} \rangle$
 - ii. Set $m' \leftarrow m - 1$ and $n' \leftarrow q^{m'}$
 - iii. Recursively decode r' using **PRS-Decoder** with input parameters $q, (k_1, k_2, \dots, k_{m'}), m'; r' \in \mathbb{F}_q^{n'}$.
3. Compute the parameter $t_m = \lceil q(1 - \sqrt{k_m/q}) \rceil$.
4. For each $m-1$ tuple $(a_1, a_2, \dots, a_{m-1})$ do:
 - i. Using Guruswami-Sudan algorithm find a list \mathcal{L}_m of codewords $c_m \in \mathcal{RS}_q(q, k_m)$ such that $d_H(c_m, \langle r_{i_m}^{a_1, a_2, \dots, a_{m-1}} \rangle) < t_m$.
 - ii. Search \mathcal{L}_m for c_m such that $d_H(c_m, \langle r_{i_m}^{a_1, a_2, \dots, a_{m-1}} \rangle)$ is least. Substitute in-place the positions corresponding to $\langle r_{i_m}^{a_1, a_2, \dots, a_{m-1}} \rangle$ with c_m .
5. **return**

OUTPUT: Resulting vector r

Pseudo Code (3)

RM-List-2

INPUT: $q, \ell \leq q, m, n \leq q^m; r = [r_1 \ r_2 \ \dots \ r_n] \in \mathbb{F}_q^n$.

STEPS:

1. For each possible m -tuple $(k_1, k_2, \dots, k_m) : k_i < q, \sum_j k_j \leq \ell$ do:
 - i. Using **PRS-Decoder** with input parameters $q, (k_1, k_2, \dots, k_m), m; r \in \mathbb{F}_q^n$, decode r as c .
 - ii. Add c to a list \mathcal{L} of codeword candidates.
2. return

OUTPUT: \mathcal{L}

Performance and Complexity

- ▶ **PRS-Decoder** can correct up to $n \prod_{i=1}^m (1 - \sqrt{\rho_i})$ errors.
- ▶ The complexity of the recursive decoder **PRS-Decoder** is $\mathcal{O}(q^{m+2})$ field operations in \mathbb{F}_q . This is $\approx \mathcal{O}(n)$ for large m .
- ▶ The complexity of **RM-List-2** is $\approx \mathcal{O}(n^2)$ field operations in \mathbb{F}_q . This is substantially better than the Pellikaan-Wu algorithm.
- ▶ Let V_m denote the volume of the rate region where the recursive algorithm performs better than the Pellikaan-Wu algorithm.

2D Performance

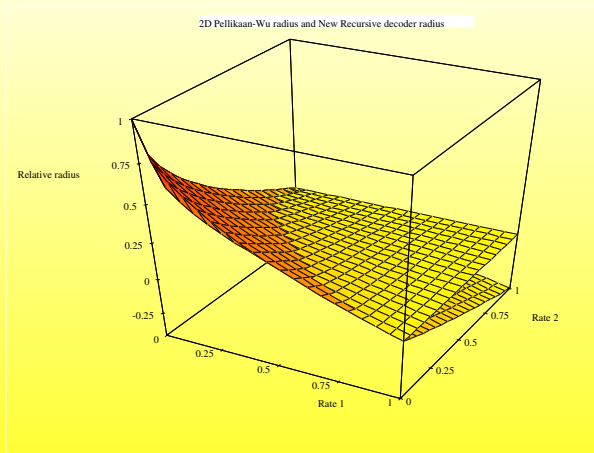


Figure: Decoding Radius of the 2D Pellikaan-Wu algorithm and the new recursive algorithm

2D Performance (contd)

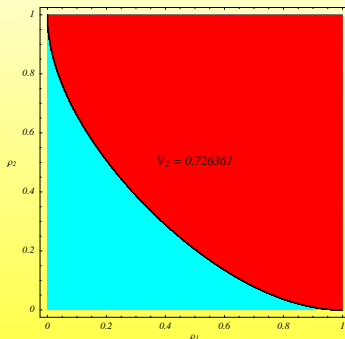


Figure: Rate Region where the 2D recursive algorithm performs better than Pellikaan-Wu algorithm

Performance of RM-List-2 for various m

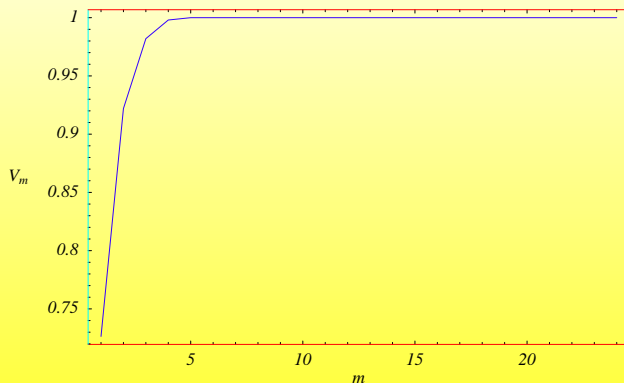






Figure: Fraction of Rate Region where the recursive algorithm performs better than Pellikaan-Wu algorithm

Conclusions

- ▶ We derived a simple correctness proof for Pellikaan-Wu algorithm.
- ▶ Obtained a constructive proof for *DeMillo-Lipton-Schwartz-Zippel* lemma.
- ▶ Proposed a recursive algebraic decoder for PRS and RM codes.
- ▶ Showed that this recursive decoder outperformed the Pellikaan-Wu decoder for much of the rate region.
- ▶ The recursive decoder is an order of magnitude more efficient than the Pellikaan-Wu algorithm and the operations are over a much smaller finite field.

References

-  R. Lidl and H. Niederreiter, *Introduction to finite fields and their applications*, University of Cambridge Press, Cambridge, 1986.
-  V. Guruswami and M. Sudan, "Improved Decoding of Reed-Solomon and Algebraic-Geometry Codes," *IEEE Trans. Inform. Theory*, **45**, No. 6, pp. 1757-1767, Sep. 1999.
-  R. Pellikaan and X.-W. Wu, "List Decoding of q -ary Reed-Muller Codes," *IEEE Trans. Inform. Theory*, **50**, No. 4, pp. 679-682, Apr. 2004.
-  R. Pellikaan and X.-W. Wu, "List Decoding of q -ary Reed-Muller Codes," Expanded version of [3], manuscript available at <http://www.win.tue.nl/~ruudp/paper/43-exp.pdf>, Nov. 2005.